

NUCLEAR CAMERA WITH  
OPEN AND FLEXIBLE SOFTWARE ARCHITECTURE

5 This invention relates to nuclear (gamma) camera  
imaging systems and, in particular, to nuclear  
cameras with software architectures which are easy to  
understand and readily adapted to changing needs of  
clinicians.

10 Traditionally, medical image data obtained from  
medical imaging systems such as nuclear cameras,  
computed tomography scanners, magnetic resonance  
imaging systems and ultrasound has been defined by  
proprietary image data formats developed by the  
various system manufacturers. Such proprietary image  
15 formats mean that the images produced by the  
different systems must be viewed or displayed by  
proprietary viewing/display applications provided by  
each vendor. A user is effectively locked into using  
the particular manufacturer's proprietary formats.  
20 Images produced by one imaging system cannot be  
readily viewed on another manufacturer's system.  
Realizing these limitations, the NEMA trade  
organization has organized and defined an image  
format which all manufacturers can use, known as the  
25 DICOM (Digital Imaging and COmmunication in Medicine)  
standard. The various manufacturers of medical  
imaging equipment can provide translators by which  
their proprietary formats can be converted into a  
common format in which users can exchange and view  
30 images on a variety of different display devices.

Data format standards such as DICOM have not  
fully solved the problems of incompatibility and ease  
of use, however. First of all, a user of a  
particular system must have access to conversion  
35 routines by which a manufacturer's proprietary image

format can be translated into the standard format.  
Conversion routines are not always readily available  
or universally successful in their operation.  
Secondly, the proprietary image data formats are  
5 typically inflexible with respect to changes.  
Changes are often desirable to enable a clinician to  
satisfy new user requirements, or incorporate  
innovations into an application, use or operation of  
the medical imaging system, or to enable the use of  
10 new software technologies that improve the  
performance of the medical imaging system. Thus,  
proprietary image formats have the effect of limiting  
innovation in imaging by clinicians and thwarting  
improvements to clinical productivity.

15 Furthermore, while image data standards such as  
DICOM enable users to exchange and use images from  
the medical imaging systems of different  
manufacturers, these standards are only effective  
when all manufacturers agree upon the standards and  
20 their requirements. The need for common agreement  
upon standards means that changes to standards cannot  
be made quickly and easily. Proposed changes need to  
be discussed, commented upon, and agreed to by the  
manufacturers before being implemented for the  
25 benefit of end users. Accordingly it is desirable  
for a medical imaging system to have a software  
architecture which overcomes these obstacles and  
readily accommodates advances in efficiencies and  
techniques as they are developed by the medical  
30 community.

In accordance with the principles of the present  
invention a nuclear camera software architecture is  
described which addresses these deficiencies of  
proprietary and standard formats. The software  
35 architecture of a camera of the present invention

allows the operation and performance of the camera to be readily modified by users. Modified data can be easily exchanged because of the self-descriptive nature of the software language and the ability to provide readily available format descriptions to all users. In a preferred embodiment the software architecture embraces an open format which is publicly available. In a constructed embodiment of the present invention the software architecture of the nuclear camera uses eXtensible Markup Language (XML) which is an open format that enables changes to the data format to be made by users and other manufacturers alike. The inventive software architecture presents data that is self-describing in that relationships between various pieces of data are captured in the format definition used to store and interpret the data. In a preferred embodiment, both image data and nuclear camera control information, in particular, study protocols, are defined in the open, extensible software architecture, enabling users to exchange not only new image data formats but also new system control procedures.

In the drawings:

FIGURE 1 illustrates in block diagram form a gamma camera constructed in accordance with the principles of the present invention;

FIGURE 2 illustrates the detector of a gamma camera in block diagram form;

FIGURE 3 illustrates a gamma camera protocol setup screen for a gated SPECT study;

FIGURE 4 illustrates the flow of control and image data in a gamma camera constructed in accordance with the principles of the present invention;

FIGURE 5 illustrates the organization of patient

data in a constructed embodiment of the present invention; and

FIGURES 6-9 illustrate the directory structure of the software architecture of a preferred embodiment of the present invention.

Referring first to FIGURE 1, a gamma camera constructed in accordance with the principles of the present invention is shown in block diagram form. A software architecture of the present invention may be used with either a single detector gamma camera, or with a dual detector gamma camera such as those shown in U.S. Patents 5,760,402 (Hug. et al.) or 6,150,662 (Hug et al.). The dual detector cameras shown in these patents are commercially available as the Forte™ and Skylight™ gamma cameras from ADAC Laboratories of Milpitas, California. These camera systems include one or more detectors 10, 12 which sense scintillation events and transfer event data over a high speed serial bus 24a, 24b from each detector to an acquisition control server 14. The acquisition control server 14 bins the event data into images, which are then sent by way of a router 20 to a database host 28 connected to a department Ethernet 26. The database host 28 is the computer on which the acquired images are normally saved. The database host is also the processing and viewing station where 3D reconstruction, re-slicing and presentation of the processed images to the user is performed. The router 20 serves to isolate internal camera data traffic from more varied departmental data traffic. A user interface personal computer 16 is coupled to the acquisition portion of the gamma camera by the router 20 and an Ethernet network 22a, 22b, 22c. The p.c. 16 controls and monitors the image data acquisition and may perform additional image

processing and display functions using an image display 18. P-scope and energy spectrum data may be displayed to the user on the display 18, enabling the user to position the patient properly in front of the camera, set energy windows correctly, and review acquired data, for instance.

A camera detector is shown in greater detail in FIGURE 2. As is well known in the art, a gamma camera detector is composed of a collimator 32, a scintillation crystal 34, and a lightguide 36. The photons produced by the crystal 34 and guided through the lightguide 36 are received by an array of photomultiplier tubes (PMTs) 38. A scintillation event is usually received over an area covering several PMTs, and the outputs of the tubes are sensed and used to locate the position on the detector at which the radiation event was received. The PMT output signals are amplified by pre-amplifiers 42 and digitally sampled by A/D converters 44. The samples from each PMT are accumulated for the duration of a scintillation event and, since multiple PMTs are involved in the detection of a single event, the accumulated outputs of multiple PMTs are accumulated to acquire the overall energy signal for a particular scintillation event. The detected energy data and location data from each scintillation event is modified by correction circuitry 46, which produces the detector outputs for energy (E) and event location (X,Y). The event data is then binned for image processing.

The p.c. 16 and user interface display 18 allow a user via keyboard and/or pointer control to select or create a predefined set of parameters (or protocols) for direction of a SPECT imaging session or other selected study by the gamma camera system.

FIGURE 3 illustrates a parameter interface screen and configurable parameters of a nuclear camera system for data acquisition that are selected and displayed on a screen by the user. FIGURE 3 illustrates some of the parameters that are configurable by the user for a desired study. It is appreciated that once set, the configurable parameters can be saved and referenced in a computer file for subsequent recall. The stored parameters or protocol file can then be recalled and utilized for another study, thus eliminating the need to again enter the parameters for similar or identical studies. The name of the parameter file shown in FIGURE 3 is "GATED SPECT" and is indicated at 300. It is appreciated that the acquisition system, once instructed by the user, will relay the parameters set by the user to the camera in order to initialize and begin a particular study. The initiation is done by selection of processing command 357. A user interface of this type is thus versatile while at the same time providing a high degree of automation of the execution of selected study protocols.

FIGURE 4 illustrates the flow of control and image data in a gamma camera constructed in accordance with the principles of the present invention. Located on the acquisition control server hardware 14 are a control server 52 and a data server 54. These servers act to access and process data stored on an xml data storage device 56. The data stored on the device 56 includes patient schedule data 56a, protocol data 56b, isotope data 56c, and collimator data 56d. This data is used to operate and control the gamma camera 82. In a preferred embodiment this data is stored in xml format as explained more fully below. The control server uses

the data stored on the device 56 and executes  
programs 52a called "scripts," also stored on the xml  
data storage device 56, to control the operation of  
the gamma camera 82. The control and data servers  
5 both operate under user control through a graphical  
user interface (GUI) client application 60 which runs  
on the user interface p.c. 16.

In operation the user defines a study (exam) by  
associating a protocol with a particular patient  
10 through the GUI client 60. The protocol comes from  
the data server 54. The GUI client then sends the  
study to the control server 52. The protocol for the  
study has a script name attached to it, enabling the  
control server to identify the script to execute  
15 within its interpreter. The script is executed to  
carry out the protocol, performing calculations as  
needed, and can obtain certain protocol parameters  
from the user such as number of azimuths, time per  
frame, and so forth. The script also prompts the  
20 user at various points during the study for further  
input as needed, such as selection of a point of  
interest to be tracked by the detectors as they  
rotate about the patient.

In a constructed embodiment the scripts are  
25 written in a scripting language called EASL. EASL  
was developed by the assignee company of the present  
invention using readily available tools for writing  
scripting language such as LEX and YACC. YACC is a  
UNIX-based tool which enables a programmer to compile  
30 the grammatical terms used in EASL to interpret the  
scripts. The EASL program is an interpreter for the  
scripts, which in turn use the xml files, and is  
stored in the control server 52.

The event data produced by the camera detectors  
35 Head1 and Head2 in this example is processed as

described above and transferred to a binner 70. The binner interacts with the control server 52 to format image data in an xml format. The formatted image data is then further processed, transferred or stored as described more fully below.

FIGURE 5 illustrates one form of organization of an xml image file. The image file includes information about the image such as the identification 90 of the patient, the nature of the nuclear study 92 which produced the image, the series 94 of the study, and the image information 96. In the illustration this information is all contained in an xml image file 100 referred to in this example as <ADACImage.xml>. The file includes the patient and study information just described, and the image data shown as 102. The file may contain other pieces of information 112, 114 which relate to the image and may also relate to each other. The meanings of these pieces of information and their relationships are captured in a definitional file called a Document Type Definition 110 (DTD) file. The image file 100 has a pointer to the DTD file 110 by which a processor or viewer can access the DTD file and interpret the information in the image file. The pointer can be to another file or directory running on the gamma camera as described below. The pointer can also be the URL of a location where the DTD file can be found. For example, if the image is being viewed by a browser, the browser will consult the DTD at the noted URL on-the-fly, and thus be able to interpret and understand the pieces 112, 114 of information which are stored in the image file 100.

An example of an <ADACImage.xml> file is shown in Appendix 1 below. It is seen that the second line of the file contains the pointer to the DTD file



ADACImage.dtd which defines the relationships of the data elements in the xml image file. The initial portion of the file contains information of the type shown at 90-94, including the equipment and detector used, the reconstruction processing used, patient information, isotope dose, and so forth. Toward the end of the file is a block of frame pixel information.

The initial portion of a DTD file which interprets the <ADACImage.xml> file is shown in Appendix 2. This <ADACImage.dtd> file is pointed to in the second line of the <ADACImage.xml> file and provides instructions to a reader as to the interpretation of the <ADACImage.xml> file. This particular DTD file begins with information as to the purpose, creation, and revision history of the DTD file. This is followed by a definition of the various elements of the image.xml file. The <ADACImage.xml> file is written in accordance with the rules set forth in the <ADACImage.dtd> file.

The xml files are seen to be written in easily understandable grammatical terms. With xml files a user is able to make changes to the xml file structure, adding user-desired information to the data already produced by the camera system. The user is able to make changes to algorithms executed by the scripts such as study protocols carried out by the gamma camera. The grammar-based nature of the xml files makes it easy to add newly defined xml files or to change existing ones, for instance, to create custom study protocols. Image files can be modified by a hospital user, for instance to add the hospital's unique codes and information such as accession numbers to image data. The DTD file referenced by an image would in that case define the

relation of the accession number to a particular image and patient. Since xml is an open format the ability to make these changes is available to all users. In most instances these changes are evident in the xml files themselves due to the self-describing nature of the xml and DTD files, with the format defining how the different elements in the data relate to each other. The xml-formatted files can be viewed on a browser or on a text editor which reveals the grammatical text of the files. The xml files can also be viewed with commercially available xml viewers such as xmlspy. The relationships of the elements of the file can be viewed in this way as the hierarchy of relationships is shown automatically by the viewer. A particular application can thus make use of all of the relationships defined for the data.

A portion of another example of an xml file, <protocol.xml>, is shown in Appendix 3. The <protocol.xml> file has a script name attached to it to enable the gamma camera to carry out a study protocol called "MIBI." Within a protocol are various steps, each representing a different acquisition step, e.g., for a static image, a stress cardiac study, or a whole body study. In the illustrated <protocol.xml> file the different characteristics of the study are called "TRAITS," which are interpreted and defined by the "protocol.dtd" file referred to by the <protocol.xml> file. Typical TRAITS include such characteristics as the detector rotation, the time at each detector orientation, the frames acquired at each step, and the isotope dosage used. The defined TRAITS are then used in the steps of the protocol identified in the file as <acquisitionstep>.

Appendix 4 illustrates an EASL script called

"tbbase.asl" for conducting a total body nuclear study. The illustrated script is written in the EASL scripting language referred to above. The script is seen to include comment lines which make the script  
5 easily understandable to users. The script is also seen to control the prompts and messages displayed to the user on the GUI as the study proceeds. In a gamma camera of the present invention a script is loaded into the interpreter and is used to run the  
10 protocols which control the operation of the camera.

Examples of the directory hierarchy of the files stored on the data storage device 56 are shown in FIGURES 6-9. In these examples the directory  
/export/home/atlas/ contains two types of data as  
15 shown in FIGURE 6. Software shipped with the gamma camera by the manufacturer is stored in the etc/ directory. This includes configuration files, default protocols, default users, default isotopes, collimators, and so forth. This is primarily data  
20 used by the control server 52, the data server 54, and the binner 70, and comprise a complete set of files necessary to operate the gamma camera. A data/ directory is also provided for storage of data created by the user. This includes acquired image  
25 data, user defined protocols, schedules of patients, isotopes, energy window settings, and other files created or modified by the user.

FIGURE 7 shows further details of the manufacturer-supplied directory etc/. The  
30 collimators.xml file contains a list of all collimators supported by the camera. Isotopes.xml and energywindowsets.xml contain lists of isotopes and their physical properties and default energy window settings. Protocols.xml contains a list of  
35 standard study protocols. Systems.xml contains a

list of systems used by the camera, and whether they are clients or servers. Users.xml contains a list of default users for the camera.

5 The directory data/ for user-created data is shown in FIGURE 8. This directory has files corresponding to those of the etc/ directory, but in this case the files are those which have been created by or modified by a user. For instance, a user may take a manufacturer-supplied protocol and modify it  
10 for a particular study which the user desires to perform. The modified protocol is stored as a protocols.xml file in the data/ directory and can be called up and executed by the user each time the user wants to conduct that study. The user-created data  
15 takes precedence over the manufacturer-supplied data. Any time a client asks for data from the data server 54, the server will first access the data/ directory to see if a user-created file satisfies the request. Only if no user-created file is found will the server  
20 look for a file in the etc/ directory.

Both directories are seen to store DTD files corresponding to the xml files, which are used to interpret the definitions in the corresponding xml files. The patient.dtd file, for instance, defines  
25 how the patient visit and study information is written out, that is, the format and relationship between the various pieces of information contained in the patients.xml file. Similarly, protocol.dtd defines how the protocol information is written out  
30 and interpreted. A DTD file is used to view an xml file when using an xml viewer such as xmlspy, or alternatively a text editor can be used to view the xml file in the form shown in the appendices.

FIGURE 9 illustrates the production and storage  
35 of an image.xml file by the camera. The binner 70

writes out the results of a particular study in xml  
format when acquisition is completed for a given  
acquisition step of a protocol. The image.xml file  
is written by the binner in accordance with the rules  
5 in the ADACImage.dtd file. This example shows an  
image file 015N0000.xml which has been stored under a  
particular patient's name, <patient\_name>/, under the  
data/Patients/ directory. Image.xml files are  
translated with the help of an "xml2peg" translator  
10 program into an image format 015N0000.img and stored  
under an x2pout/ directory. This directory is  
accessed by the database host 28 in response to an  
external user request for the patient's images. When  
a particular image has been exported to the database  
15 host, the image is then stored under the  
SentToDatabase/ directory and a confirmation of the  
export is sent as a status update to the data server  
54. Thus, images are available in both an xml format  
and in an image format which is most efficient for  
20 storage, transport or display by the database host.

## Appendix 1 - <ADACImage.xml>

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE ADACImage SYSTEM "adacimage.dtd">

5  <ADACImage>
    <ADACPrivateIOD>
        <ADACAcquisitionIE numberOfXYSets="1" objectDataType="SP"
10  uniqueObjectKey="01" uniquePatientKey="" corCorrectionFlag=""
    scaleFactorOfSets="1.0" isotopeImagingMode="0"
    totalAcquisitionTime="60000" uniformityCorrection="">
        </ADACAcquisitionIE>
        <ADACEquipmentIE clientName="cameraclient">
            <ADACDetectorModule fieldOfViewInMM="597.000000"
15  leadingDetector="head1" calibrationFactor="0.145752"
            crystallThickness="0.000000" tubePatternCorection="180"
            axisOfRotaionCorrection="0" angleBetweenTheDetectors="180"/>
        </ADACEquipmentIE>
        <ADACReconstructionAndProcessingIE manipulatedImage="N"
20  curveOrROISetName="" specificCurveName="" reconstructionType=""
        associatedParentFile="" associatedNormalCurveFile=""
        associatedPathAndFileName="" xDimensionOfAssciatedImage="0"
        yDimensionOfAssciatedImage="0" approximateEjectionFraction="0.0"
        associatedHistogramCurveFile=""
25  limitedReconstructionStartSlice="0">
            <FilterModule filterType="" filterOrder="0"
            filterCutoffFrequcey="0.0"/>
            <ROIModule roiMode="" roiName="" roiType="" roiColor=""
            associatedROI_ID="" roiNumberOfPoints="" centerOfBoundingBox=""
30  roiBoundingBoxCoords="" xLengthOfBoundingBox=""
            yLengthOfBoundingBox="" imageFrameNumberOfROI=""
            numberOfCurvesOrROIsInSet=""/>
            <ReorientationModule reorientationType=""
            reorientationAzimuth="0" reorientationElevation="0"/>
35  <ColorModule trueColorFlag="1" associatedColorMap="0"
            customizedColorMap="" lowerLevelGrayShade="0"
            upperWindowGrayShade="0"/>
        </ADACReconstructionAndProcessingIE>
        <ADACMiscIE unUsed1="" unUsed2="" unUsed3="" unUsed4=""
40  dirEntries="" maxTimeValue="" minTimeValue="" vfrStructure=""
        rateDifference="" imageOrientation="0" sizeOfSubHeaders=""
        binningAlgoVersion="" dataStructureArray=""
        adacPatientPosition="other" programSpecifiResult=""
        verticalFOVOffsetInMM="0.000000"
45  horizontalFOVOffsetInMM="0.000000">
        </ADACMiscIE>
        <ADACCorrectionsIE>
            <CORCorrectionModule>
            </CORCorrectionModule>
50  <ImageOffsetCorrectionModule>
            </ImageOffsetCorrectionModule>
        </ADACCorrectionsIE>
    </ADACPrivateIOD>
    <NMImageIOD>
55  <PatientIE>
```

```

    <PatientModule patientsID="1122" ethnicGroup=""
patientsSex="0" patientsName="First_da26, Gary"
patientComments="" patientsBirthDate="" patientsBirthTime="">
    </PatientModule>
5    </PatientIE>
    <StudyIE>
        <GeneralStudyModule studyID="015700" studyDate="20010507"
studyTime="13:25:51" accessionNumber="::200157D0J05"
studyInstanceID="" studyDescription="BASE-static"
10    referringPhysiciansName="">
            <ProcedureCodeSequence>
                <CodeSequenceMacro>
                    <BasicCodedEntry/>
                    <EnhancedEncodingMode/>
15                </CodeSequenceMacro>
            </ProcedureCodeSequence>
        </GeneralStudyModule>
        <PatientStudyModule occupation="" patientsAge="0"
patientsSize="0" patientsWeight="0" additionalPatientsHistory=""
20    admittingDiagnosesDescription=""/>
            </StudyIE>
            <SeriesIE>
                <GeneralSeriesModule modality="NM" seriesDate="20010507"
seriesTime="13:25:51" protocolName="" seriesNumber=""
25    operatorsName="" patientPosition="OTHER" seriesDescription=""
seriesInstanceUID="" performedProcedureStepID=""
performingPhysiciansName="" largestPixelValueInSeries="2.000000"
smallestPixelValueInSeries="0.000000"
performedProcedureStepStartDate=""
30    performedProcedureStepStartTime=""
performedProcedureStepDescription="">
                    </GeneralSeriesModule>
                </SeriesIE>
                <FrameOfReferenceIE>
                    <FrameOfReferenceModule frameOfReferenceUID=""
positionReferenceIndicator=""/>
                    </FrameOfReferenceIE>
                    <EquipmentIE>
                        <GeneralEquipmentModule stationName="Atlantis"
40    manufacturer="ADAC Laboratories" institutionName="ADAC01"
pixelPaddingValue="" spatialResolution="1.0"
deviceSerialNumber="atlas-97001" dateOfLastCalibration=""
timeOfLastCalibration="" manufacturersModelName=""
institutionalDepartmentName="">
                            <InstitutionAddress>
                                <Address zip="" city="" state="" country=""
streetName="" numberInTheStreet=""/>
                                </InstitutionAddress>
                                <SoftwareVersions binnerVersion="8.0b"
50    dataServerVersion="" controlServerVersion=""
graphicUserInterfaceVersion=""/>
                                    </GeneralEquipmentModule>
                                </EquipmentIE>
                                <ImageIE>
35    <GeneralImageModule imageDate="" imageTime=""
imageType="STATIC" imageComments="" instanceNumber=""
acquisitionDate="20010507" acquisitionTime="13:25:51"

```

```
acquisitionNumber="" patientOrientation=""
imagesInAcquisition="" qualityControlImage="N"
derivationDescription="" lossyImageCompressionRatio="">
  </GeneralImageModule>
5    <NMImagePixelModule highBit="11" bitsStored="12"
    pixelSpacing="" bitsAllocated="16" samplesPerPixel="1"
    photometricInterpretation="PALETTECOLOR"/>
    <MultiFrameModule numberOfFrames="2">
      <FrameIncrementPointer>
10      <DetectorVector> 1, 2</DetectorVector>
      </FrameIncrementPointer>
    </MultiFrameModule>
    <NMMultiFrameModule numberOfPhases="NO_VALUE_SET"
    numberOfSlices="" numberOfDetectors="2"
15    numberOfRotations="NO_VALUE_SET"
    numberOfTimeSlots="NO_VALUE_SET"
    numberOfFRRIntervals="NO_VALUE_SET"
    numberOfEnergyWindows="NO_VALUE_SET">
      <FrameIncrementPointer>
20      <DetectorVector> 1, 2</DetectorVector>
      </FrameIncrementPointer>
    </NMMultiFrameModule>
    <NMImageModule imageID="" countRate="" imageType="STATIC"
    tableHeight="0.000000" tableTraverse=""
25    countsAccumulated="766.000000" processingFunction=""
    actualFrameDuration="30000" triggerSourceOrType=""
    lossyImageCompression="00">
      </NMImageModule>
      <NMIsotopeModule>
30      <EnergyWindowInformationSequence>
        <EnergyWindowRangeSequence/>
      </EnergyWindowInformationSequence>
      <RadiopharmaceuticalInformationSequence
    radiopharmaceutical="" radionuclideTotalDose="0.000000"
35    radiopharmaceuticalRoute="" radiopharmaceuticalVolume=""
    radiopharmaceuticalStopTime="" radiopharmaceuticalStartTime="">
        <CalibrationDataSequence syringeCounts=""
    energyWindowNumber="" residualSyringeCounts=""/>
      </RadiopharmaceuticalInformationSequence>
40    <RadiopharmaceuticalInformationSequence
    radiopharmaceutical="" radionuclideTotalDose="0.000000"
    radiopharmaceuticalRoute="" radiopharmaceuticalVolume=""
    radiopharmaceuticalStopTime="" radiopharmaceuticalStartTime="">
        <CalibrationDataSequence syringeCounts=""
45    energyWindowNumber="" residualSyringeCounts=""/>
      </RadiopharmaceuticalInformationSequence>
      <InterventionDrugInformationSequence
    interventionDrugDose="" interventionDrugName=""
    interventionDrugStopTime="" interventionDrugStartTime="">
50    </InterventionDrugInformationSequence>
      </NMIsotopeModule>
      <NMDetectorModule>
        <DetectorInformationSequence startAngle="0"
    zoomCenter="" zoomFactor="0.010000" focusCenterX=""
55    focusCenterY="" focalDistance="200.000000" imagePosition=""
    collimatorType="NONE" radialPosition="0.000000"
    imageOrientation="" gantryDetectorTilt=""
```



```

collimatorOrGridName="INTR" centerOfRotationOffset=""
distanceSourceToDetector="">
    <HexagonalDimension/>
    </DetectorInformationSequence>
5    </NMDetectorModule>
    <NMReconstructionModule sliceLocation=""
sliceThickness="10" convolutionKernel="" spacingBetweenSlices=""
reconstructionDiameter=""/>
    <ImagePixelModule rows="256" columns="256" highBit="11"
10    bitsStored="12" bitsAllocated="16" samplesPerPixel="1"
pixelAspectRatio="1" pixelRepresentation="0000H"
largestImagePixelValue="" redPaletteColorLUTData=""
bluePaletteColorLUTData="" smallestImagePixelValue=""
greenPaletteColorLUTData=""
15    photometricInterpretation="PALETTECOLOR"
redPaletteColorLUTDescriptor="" bluePaletteColorLUTDescriptor=""
greenPaletteColorLUTDescriptor="">
    <Frame gateNumber="0" frameNumber="1"
imageViewID="View03.Det1" timePerFrame="30000"
20    countsPerFrame="766" detectorNumber="1" maxPixelOfFrame="2"
minPixelOfFrame="0">
iVBORw0KGgoAAAANSUHEUgAAQAAAAEAEAAAAAApiSv5AAAGt0lEQVR4nO3
d25bbNgyFYbl5/3fuTVczY1MWDyCwAf7fRVamtSQQpHWgwMl1AQAAAAA
AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAMC716bPWm+NY6wM1NfTGZQHo
25    tMn5U5c7uhdkCJ0sBkmT3v5/P8+w/OILOFPI49IBIBg+840xc9h35q3t+mv
jz/Ht8YS3STGRaabby3umms9rXtYWhvIgCHPZNJxBdCJbkjloTJ2vH3MI3v
8x/zoRnS68tX8aTU+nfYFIxG22sNVVtz0Ty77SkoSZTlRqA9iW+J+9DodV0
Pq/ribsU/dqAUF3l2sX928miSXulPV6Ij7sliN9k05uOmBEmbdOuSEKcikW
nqrtdQdt04Xg/cf/Rdg014EL9u/p6PX4FK/158n5smYAf4w91QtZylsMhE
30    3z425nzlFBhBIbaXRBtufn+PDrrz+NVjm4fX1b0HCA7ikOmGiRismHd34S+C
KfwbvOob3m0w68JfTlmN5/sYRR8KhJZAke0q/1Ek/ZX6zAMP+7NntdVORvz
hBfyg8S9UGxceusUhsa2+TVfKq6U+Y7CKIaTODRWyAtcPxC/L5SLb33GLp9
0Gx01/H5aB/2uLuUztT5n/D+fzJ44bIs9jXqVF71ziig0xP9757r3fT+kh9
hPsVko3uY6ww7f3Th9YfXJf+kGuLuF8yoRKgShzPhKeD8dfBp9Bnucy8Z0
35    WH4FGDD/MhVpt1Q4G7LshGq/70XpK2LGvNBQbvWOoti+w3/eomJsOn8+un3
9P2rROaVt3H/jooqbr7T1TFnes8SOHDiWlcawUy/F4dw55s0MybrnGyAvmg
1+41VmFlUpBu8/u5hiLURP3YonWrdF/f7Zm0KNrBPsw8+hm5aq7cZrEauZh
362vyMASULiMGDWGgf/3OWXe9gStNE84Qz0+aavXf5LvJCLRtqdThaFywHo
C3+ZlndL5Lk+koloJHC8aWf/s7BZ2W5sRCWPAroJTFngP02ps75OCZrHXPI
40    0o5FSuxGvHoluzHUat4zDAuvP2VhbvcEZ1pLHoZK3fAEa3oFOWK6z9vltX2
pjp6QxXpc08u7jwIFSq8m1WjPk4KbXkmw5wxulWGYkmnm8e9N1f0XLNks+
MhmTvcahuLH0f3b8fPcV73iN5s2uF7jft5T4CEzYvIodTcb753WTqRvZgmq
/jef5ml6rvR9yNy8y+mSZi6qu80iT7TH8OzbZUOplVawlu4WttHXMWmHZVe
n5FVJ43AVo9dJ/tAvG17aBiLzfrbLDzrpmICZRRWnqOctP2S5+F3+tnZvhs
45    5oXvNtL9t1GDGoPAKtiEBF9YWq+8UqS4uu6jq/92evnRUKjuVQCD7NI1fq3
jA57o3Xi8rv6xrc1jVzF4DPuapu+R6cR+zYZZu7bRyzeMzr6Zg56Z9hHXin
d/Ww9Y/BEdtjJBnaprlqyeTehnPevnr/LOk0bm7bp2dcuOjkrohdhMHTJIOZ
hWgt7/8De3Tmb1z1+jjCzhUNJI3Nj+VvwPgQkJh1FciGrJUy1nexyaRdp2p
aKzW/YXg6gMFSe2pd3zx4KZ+zTvKP7S7FSLm8KUS8as38KpfPbbvHzCVJpK
50    1f5UwQ6o2i4HpK5XT6a2ZXP9HZXmE4TPUuuvE/jWuBy3OhliRFozJ27ZIRn
z8nR/OvQKQ6lT3kq15lBrj12H7F3EcNZYPKulpqJTp3DyL74yaG65iE4yFI
bIAWKGRvx0MoPj8kwCJaDLLKcj75ZRdznpn+njBfEW7GLKTIIdLOV3ettfN
9z+WaUrVOJoEA5NORZD0AU2xs9Kbpl/OhD/dKm01F9Hn2qBXUeJe4IpTOPc
otAlgzlQaP6zkQVc65VNVtvsuRfRLyOFD3uaIp4gXWuJKzXnJS1v3tpVhP5
55    ng912Ry3yGtnr8PP38PHDZ3Y5TAnlGnRdl+8qqfvPlcxtP5nGZy3u0Hgk/m
ruISxN0bSMkUfepKye1FVfUCerWlgr2rhfMeShVgVzjig38JpD8L3VSnBV/
y064NVZAu9Zhm0/r71qzcBbqNmqqoqlufsSjNl08G2AnjkWJ//TasMNXHX

```

```
toP3Xm6hn8L+aeyfn9P756228KkFijx6SZbTGVad4HmZwZuz0l1/8coS5XC
/x+b3atv3eGK8bxKRzvxT9a77gOc97bu3OVb09LMNhsEP3u//ReC0N3Xdd
mdiElnQudl2nktXtAuCa+ibssMeS0xsTia3T5ghu6Akb+PmQwgAAAAAAAAA
5  AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAE7+BWA6AwG7AX2YAAAAAE1F
TkSuQmCC</Frame>
    <Frame gateNumber="0" frameNumber="2"
imageViewID="View04.Det2" timePerFrame="30000"
countsPerFrame="0" detectorNumber="2" maxPixelOfFrame="0"
minPixelOfFrame="0">
10  iVBORw0KGgoAAAANSUhEUgAAQAAAEAEAAAAAapiSv5AAAA1U1EQVR4nO3
BAQEAAACakP6v7ggKAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
AAAAAAAAAgKoBAR4AAZF/okMAAAAASUVORK5CYII=</Frame>
15  </ImagePixelModule>
    </ImageIE>
    </NMImageIOD>
</ADACImage>
```

## Appendix 2 - <ADACImage.dtd>

```

<!--
5   file :   ADACImage.dtd
   purpose:  This is the DTD used to generate the XML image from the Binner
   created:  26-Oct-00
   property of: ADAC Laboratories

   revision history:
10   26-Oct-00  CM    initial version
   21-Nov-00  CM    i) Changed MultiframeModule to MultiFrameModule
                   ii) The pixelData is moved in to the Element.
                   iii) The order of the elements are changed so that
15   all the encoded data will appear at the very end
                   of the XML image
   04-Jan-01  CM    i) Removed the Variable Frame Element
                   ii) Added the gateNumber in Frame Element
                   iii) Added the detectorNumber in Frame Element
20   25-Jan-01  CM    i) Changes which will make the DTD less restrictive to
                   generate XML output and validation are made. These
                   include changing mandatory Elements to optional etc.
                   ii) ID's were replaced by CDATA
   15-Feb-01  CM    Added the Element ADACCorrectionsIE and related elements
   16-Feb-01  CM    Added the Element NameAndValue to hold data from control server
25   04-Mar-01  CM    Following the fixes in the xmllib the ADACCorrectionsIE and
                   other elements were changed as optional from required.
   30-Mar-01  CM    Added imageOrientation,patientRoll and adacPatientPosition
-->
30   <!ELEMENT ADACImage (ADACPrivateIOD, NMImageIOD)>
   <!ELEMENT ADACPrivateIOD (ADACAcquisitionIE, ADACEquipmentIE,
   ADACReconstructionAndProcessingIE, ADACMiscIE, ADACCorrectionsIE)>
   <!ELEMENT NMImageIOD (PatientIE, StudyIE, SeriesIE, FrameOfReferenceIE?, EquipmentIE,
   GeneralIE?, ImageIE)>
35   <!ELEMENT ADACAcquisitionIE (StaticAcquisitionModule?, DynamicAcquisitionModule?,
   GatedAcquisitionModule?, MCDAcquisitionModule?, TotalBodyAcquisitionModule?,
   VANTAGEAcquisitionModule?)>
   <!-- ATTLLIST ADACAcquisitionIE
   isotopeImagingMode CDATA #IMPLIED
   uniformityCorrection CDATA #IMPLIED
40   totalAcquisitionTime CDATA #IMPLIED
   objectDataType CDATA #IMPLIED
   uniquePatientKey CDATA #IMPLIED
   uniqueObjectKey CDATA #IMPLIED
   corCorrectionFlag CDATA #IMPLIED
45   numberOfXYSets CDATA #IMPLIED
   scaleFactorOfSets CDATA #IMPLIED
   -->
   <!ELEMENT ADACEquipmentIE (ADACDetectorModule)>
   <!-- ATTLLIST ADACEquipmentIE
50   clientName CDATA #IMPLIED
   -->
   <!ELEMENT ADACReconstructionAndProcessingIE (FilterModule, ROIModule, ReorientationModule,
   ColorModule)>
   <!-- ATTLLIST ADACReconstructionAndProcessingIE
55   associatedParentFile CDATA #IMPLIED
   associatedHistogramCurveFile CDATA #IMPLIED
   associatedNormalCurveFile CDATA #IMPLIED
   associatedPathAndFileName CDATA #IMPLIED
   reconstructionType CDATA #IMPLIED
60   limitedReconstructionStartSlice CDATA #IMPLIED

```

```
manipulatedImage CDATA #IMPLIED
curveOrROISetName CDATA #IMPLIED
specificCurveName CDATA #IMPLIED
5 xDimensionOfAssociatedImage CDATA #IMPLIED
yDimensionOfAssociatedImage CDATA #IMPLIED
approximateEjectionFraction CDATA #IMPLIED
imageViewID CDATA #REQUIRED
>
<!ELEMENT ADACMiscIE (NameAndValue*)>
10 <!ATTLIST ADACMiscIE
sizeOfSubHeaders CDATA #IMPLIED
dirEntries CDATA #IMPLIED
minTimeValue CDATA #IMPLIED
maxTimeValue CDATA #IMPLIED
15 rateDifference CDATA #IMPLIED
dataStructureArray CDATA #IMPLIED
vfrStructure CDATA #IMPLIED
horizontalFOVOffsetInMM CDATA #IMPLIED
verticalFOVOffsetInMM CDATA #IMPLIED
20 unUsed1 CDATA #IMPLIED
unUsed2 CDATA #IMPLIED
unUsed3 CDATA #IMPLIED
unUsed4 CDATA #IMPLIED
binningAlgoVersion CDATA #IMPLIED
25 programSpecifiResult CDATA #IMPLIED
imageOrientation CDATA #IMPLIED
patientRoll CDATA #IMPLIED
adacPatientPosition CDATA #IMPLIED
>
30 <!ELEMENT ADACCorrectionsIE (CORCorrectionModule, ImageOffsetCorrectionModule)>
<!ELEMENT PatientIE (PatientModule)>
<!ELEMENT StudyIE (GeneralStudyModule, PatientStudyModule?)>
<!ELEMENT SeriesIE (GeneralSeriesModule, NMPETPatientOrientationModule?)>
<!ELEMENT FrameOfReferenceIE (FrameOfReferenceModule)>
35 <!ELEMENT EquipmentIE (GeneralEquipmentModule)>
<!ELEMENT GeneralIE (VOI_LUTModule?, SOPCommonModule?)>
<!ELEMENT ImageIE (GeneralImageModule, NMImagePixelModule, MultiFrameModule,
NMMultiFrameModule, NMImageModule, NMIsotopeModule, NMDetectorModule,
40 NMTomoAcquisitionModule?, NMMultiGatedAcquisitionModule?, NMPhaseModule?,
NMReconstructionModule?, OverlayPlaneModule?, MultiframeOverlayModule?, CurveModule?,
ImagePixelModule)>
<!ELEMENT PatientModule (ReferencedPatientSequence?, OtherPatientID*, OtherPatientName*)>
<!ATTLIST PatientModule
45 patientsName CDATA #REQUIRED
patientsID CDATA #REQUIRED
patientsBirthDate CDATA #REQUIRED
patientsSex (M | F | O) #REQUIRED
patientsBirthTime CDATA #IMPLIED
ethnicGroup CDATA #IMPLIED
50 patientComments CDATA #IMPLIED
>
<!ELEMENT GeneralStudyModule (PhysicianOfRecord*, NameOfPhysicianReadingStudy*,
ReferencedStudySequence*, ProcedureCodeSequence+)>
<!ATTLIST GeneralStudyModule
55 studyInstanceID CDATA #REQUIRED
studyDate CDATA #REQUIRED
studyTime CDATA #REQUIRED
referringPhysiciansName CDATA #REQUIRED
studyID CDATA #REQUIRED
60 accessionNumber CDATA #REQUIRED
studyDescription CDATA #IMPLIED
>
```

```

<!ELEMENT PatientStudyModule EMPTY>
<!--ATTLIST PatientStudyModule
  admittingDiagnosesDescription CDATA #IMPLIED
  patientsAge CDATA #IMPLIED
  patientsSize CDATA #IMPLIED
  patientsWeight CDATA #IMPLIED
  occupation CDATA #IMPLIED
  additionalPatientsHistory CDATA #IMPLIED
-->
<!--ELEMENT GeneralSeriesModule (ReferencedStudyComponentSequence?,
RequestAttributeSequence*, PerformedActionItemSequence*)-->
<!--ATTLIST GeneralSeriesModule
  modality (CR | MR | US | BI | DD | ES | MA | PT | ST | XA | RTIMAGE | RTSTRUCT | RTRECORD |
  DX | IO | GM | XC | CT | NM | OT | CD | DG | LS | MS | RG | TG | RF | RTDOSE | RTPLAN | HC | MG |
  PX | SM) #REQUIRED
  seriesInstanceUID CDATA #REQUIRED
  seriesNumber CDATA #REQUIRED
  laterality (R | L) #IMPLIED
  seriesDate CDATA #IMPLIED
  seriesTime CDATA #IMPLIED
  performingPhysiciansName CDATA #IMPLIED
  protocolName CDATA #IMPLIED
  seriesDescription CDATA #IMPLIED
  operatorsName CDATA #IMPLIED
  bodyPartExamined (SKULL | CSPINE | TSPINE | LSPINE | SSPINE | COCCYX | CHEST | CLAVICLE
  | BREAST | ABDOMEN | PELVIS | HIP | SHOULDER | ELBOW | KNEE | ANKLE | HAND | FOOT |
  EXTREMITY | HEAD | HEART | NECK | LEG | ARM | JAW) #IMPLIED
  patientPosition (HFP | HFDR | FFDR | FFP | HFS | HFDL | FFDL | FFS | OTHER) #IMPLIED
  smallestPixelValueInSeries CDATA #IMPLIED
  largestPixelValueInSeries CDATA #IMPLIED
  performedProcedureStepID CDATA #IMPLIED
  performedProcedureStepStartDate CDATA #IMPLIED
  performedProcedureStepStartTime CDATA #IMPLIED
  performedProcedureStepDescription CDATA #IMPLIED
-->

```

### Appendix 3 - <protocol.xml>

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE protocols SYSTEM "protocol.dtd">
5 <protocols>
  <!-- protocol record # 0 -->
  <protocol>
    <protoinfo>
      <protoname type="TRAIT_PROT_NAME">MIBI MIBI</protoname>
10    <tval type="TRAIT_PROT_ANATOMICAL">Cardiac</tval>
    </protoinfo>
    <protosteps>
      <protostep>
        <acquisitionstep>
15          <acqstepinfo>
            <tval type="TRAIT_ACQR_SCRIPT_NAME">ect</tval>
          </acqstepinfo>
          <acqstepaction>
            <concurrent>
20              <tval type="TRAIT_ACQR_NAME">Rest MIBI</tval>
              <tval
type="TRAIT_ACQR_APPDATA_CLASSNAME">data.SPECTAcqStep</tval>
                <tval type="TRAIT_ACQR_X_DIMENSION">64</tval>
                <tval type="TRAIT_ACQR_Y_DIMENSION">64</tval>
25                <tval type="TRAIT_ACQR_ROTATION_CLOCKWISE">true</tval>
                <tval type="TRAIT_ACQR_TOTAL_ROTATION">180</tval>
                <tval type="TRAIT_ACQR_START_ANGLE">0</tval>
                <tval type="TRAIT_ACQR_RELATIVE_ANGLE">90</tval>
                <tval type="TRAIT_ACQR_DEPTH">16</tval>
30                <tval type="TRAIT_ACQR_MAX_PIXEL">65535</tval>
                <tval type="TRAIT_ACQR_ZOOM">1</tval>
                <tval type="TRAIT_ACQR_DETECTORS">12</tval>
                <tval type="TRAIT_ACQR_FLOOD_TABLE">INTR.FCR</tval>
                <tval type="TRAIT_ACQR_MILLISECONDS">10000</tval>
35                <tval type="TRAIT_ACQR_KCOUNTS">0</tval>
                <tval type="TRAIT_ACQR_IS_GATED">>false</tval>
                <tval type="TRAIT_ACQR_STOP_ON_SATURATE">>false</tval>
                <tval type="TRAIT_ACQR_IS_COINCIDENT">>false</tval>
                <tval type="TRAIT_ACQR_FRAMES_PER_AZIMUTH">1</tval>
40                <tval type="TRAIT_ACQR_NUM_AZIMUTHS">32</tval>
                <tval type="TRAIT_ACQR_COMBINE_HEADS">true</tval>
                <tval type="TRAIT_ACQR_IS_ECT_CONTINUOUS">>false</tval>
                <tval type="TRAIT_ACQR_DOSAGE_ADMIN_TIME">00:00:00</tval>
                <tval type="TRAIT_ACQR_IS_ECT_NONCIRCULAR">true</tval>
45              </concurrent>
            </acqstepaction>
          </acquisitionstep>
        </protostep>
      <protostep>
50        <acquisitionstep>
          <acqstepinfo>
            <tval type="TRAIT_ACQR_SCRIPT_NAME">ect</tval>
          </acqstepinfo>
          <acqstepaction>
            <concurrent>
55              <tval type="TRAIT_ACQR_NAME">Stress MIBI</tval>
              <tval
type="TRAIT_ACQR_APPDATA_CLASSNAME">data.SPECTAcqStep</tval>
                <tval type="TRAIT_ACQR_X_DIMENSION">64</tval>
60                <tval type="TRAIT_ACQR_Y_DIMENSION">64</tval>
                <tval type="TRAIT_ACQR_ROTATION_CLOCKWISE">true</tval>
```

```

5      <tval type="TRAIT_ACQR_TOTAL_ROTATION">180</tval>
      <tval type="TRAIT_ACQR_START_ANGLE">0</tval>
      <tval type="TRAIT_ACQR_RELATIVE_ANGLE">90</tval>
      <tval type="TRAIT_ACQR_DEPTH">16</tval>
      <tval type="TRAIT_ACQR_MAX_PIXEL">65535</tval>
      <tval type="TRAIT_ACQR_ZOOM">1</tval>
      <tval type="TRAIT_ACQR_DETECTORS">12</tval>
      <tval type="TRAIT_ACQR_FLOOD_TABLE">INTR.FCR</tval>
10     <tval type="TRAIT_ACQR_MILLISECONDS">10000</tval>
      <tval type="TRAIT_ACQR_KCOUNTS">0</tval>
      <tval type="TRAIT_ACQR_IS_GATED">>false</tval>
      <tval type="TRAIT_ACQR_STOP_ON SATURATE">>false</tval>
      <tval type="TRAIT_ACQR_IS_COINCIDENT">>false</tval>
15     <tval type="TRAIT_ACQR_FRAMES_PER_AZIMUTH">1</tval>
      <tval type="TRAIT_ACQR_NUM_AZIMUTHS">32</tval>
      <tval type="TRAIT_ACQR_COMBINE_HEADS">>true</tval>
      <tval type="TRAIT_ACQR_IS_ECT_CONTINUOUS">>false</tval>
      <tval type="TRAIT_ACQR_DOSAGE_ADMIN_TIME">00:00:00</tval>
      <tval type="TRAIT_ACQR_IS_ECT_NONCIRCULAR">>true</tval>
20     </concurrent>
      </acqstepaction>
      </acquisitionstep>
      </protostep>
      </protosteps>
25    </protocol>
    <!-- protocol record # 1 -->
    <protocol>
      <protoinfo>
        <protoname type="TRAIT_PROT_NAME">Dual Isotope</protoname>
        <tval type="TRAIT_PROT_ANATOMICAL">Cardiac</tval>
30      </protoinfo>
      <protosteps>
        <protostep>
          <acquisitionstep>
            <acqstepinfo>
              <tval type="TRAIT_ACQR_SCRIPT_NAME">ect</tval>
35            </acqstepinfo>
            <acqstepaction>
              <concurrent>
                <tval type="TRAIT_ACQR_NAME">ect</tval>
40                <tval

```

#### Appendix 4 - tbbase.asl

```
#
# file: tbbase.asl
5 # purpose: PreProgram Script for 'basic' TotalBody(tb) study.
#
# created: 3-Jan-01
# property of ADAC Laboratories
#
10 # revision history:
#     3-Jan-00      DC      initial version
#
# script dependencies:
#     - Exchanger is parked (check ColPark)
15 #     - User Override is Off
#     - Rings are locked at 180.
#
# target values (Input):
#     none
20 #
# return values (Output):
#     false - Script failed to complete
#     true - Script completed
#
25 # $Header: /usr/adac/Repository/camera/camera/control/etc/easl.scripts/tbbase.asl,v 1.18
2001/05/08 05:58:15 sheavner Exp $
#
script "tbbase"

30     # Debug message - let me know the script has started ok
        choiceNum = message( "DEBUG MESSAGE -- Starting tbbase.asl - Gantry will move
to scan setup position when you click ok")
        choices "DEBUG OK!";

35     # Discuss study with tech, subtract off one FOV
        if ( TRAIT_ACQR_SCAN_LENGTH > 0 ) {
            TRAIT_ACQR_SCAN_LENGTH = TRAIT_ACQR_SCAN_LENGTH -
TRAIT_ACQR_FOV_SIZE_Y;
        } else {
40             TRAIT_ACQR_SCAN_LENGTH = TRAIT_ACQR_SCAN_LENGTH +
TRAIT_ACQR_FOV_SIZE_Y;
        }

        TABLE_EXTEND_POSITION_VELOCITY = 40.0;

45     #
        # Set the starting and ending(target) position of the longitudinal
        # motion and calculate the scan length, in microns for the binner.
        #
50     _startTablePos = TRAIT_ACQR_TRANSLATE_POS;
        _endTablePos = TRAIT_ACQR_TRANSLATE_POS +
TRAIT_ACQR_SCAN_LENGTH;

55     _tb_lower_limit = GantryTB lower limit;
```



```

    _tb_upper_limit = GantryTB_upper_limit;
    if ( (_startTablePos > _tb_upper_limit) or ( _startTablePos < _tb_lower_limit) ) {
        choiceNum = message("EASL_STUDY_TB_START_POS",
    5  _tb_lower_limit/10, _tb_upper_limit/10, ( _tb_upper_limit-
        _tb_lower_limit+TRAIT_ACQR_FOV_SIZE_Y)/10 ) choices "EASL_CONTINUE";
        return;
    }
    if ( (_endTablePos > _tb_upper_limit) or ( _endTablePos < _tb_lower_limit) ) {
    10  if (TRAIT_ACQR_SCAN_LENGTH > 0) {
        _allowable_length = _tb_upper_limit-
        _startTablePos+TRAIT_ACQR_FOV_SIZE_Y;
    } else {
        _allowable_length = _tb_lower_limit-
    15  _startTablePos+TRAIT_ACQR_FOV_SIZE_Y;
    }
    choiceNum = message("EASL_STUDY_TB_END_POS",
        _tb_lower_limit/10, _tb_upper_limit/10, _allowable_length/10, _startTablePos/10 ) choices
    "EASL_CONTINUE";
    return;
    }

    #
    # Negative scan length implies that we are scanning from foot to head
    #

    25  show status " ***** (script)           Starting position is " + _startTablePos + "
mm";
    show status " ***** (script)           Ending(target) position is " +
        _endTablePos + " mm";
    30  show status " ***** (script)           The required scan length is " +
        TRAIT_ACQR_SCAN_LENGTH + " mm";

    # Leftover -- remove when can verify it's ok
    StudyVGAngle = 180; start StudyVGAngle;

    35  # calculate the longitudinal velocity
    # ( negative indicates that we are moving from foot to head,
    #     the start position is greater than the stop position )

    40  # _scanSpeed is mm per second
    _scanSpeed = TRAIT_ACQR_TBODY_SPEED;

    # test the speed for a valid range

    45  show status " ***** (script)           Scan Speed is " + _scanSpeed + "
mm/sec";

    # -----
    #

    50  show status "%%%%%%%%%(script)           Running TB setup script
%%%%%%%%";
    _finishedSetup = false;
    while ( _finishedSetup = false)
    {
    55  StudyTBSetup = _startTablePos;
        run StudyTBSetup;
    }
```

```

        if (StudyTBSetup != true)
        {
            if (StudyTBSetup < 0)
                return;
5         choicenum = message("EASL_STUDY_SETUP_FAILED") choices
"EASL_RETRY", "EASL_CANCEL";
            if (choicenum = 2)
                return;
        }
10        else
        {
            _finishedSetup = true;
        }
    }
15

# -----
#

20        # Start binner and wait for the start button
        show status "%%%%%%%%%(script)      WAITING FOR START
%%%%%%%%";

        choiceNum = message( "DEBUG MESSAGE -- Gantry is in position, Click start to
25        begin study (after clicking ok)" )
            choices "OK";
        # status message( "Press the START button to begin");

        enable imaging;
30        wait until user requests imaging start;

        show status "%%%%%%%%%(script)      User has pressed START
%%%%%%%%";

35        # -----
        #
        #
        # For Total Body acquisitions there are 3 stages
        #
40        # 1. The mask is opened one row at a time and when fully open the binning halts
        waiting for the table to start moving
        # 2. The table moves the required distance at the required speed with the mask fully
        open and binning in progress
        # 3. The table stops moving and the mask automatically starts to close.
45        #

        #
        # Setup the target position for the acquisition and the speed for the table motion during
        the acquisition
50        #
        StudyTB = _endTablePos;
        StudyTB velocity = _scanSpeed;

        # Start StudyMonitor to watch for gantry exceptions, start after setup.ppm completes
55        start StudyMonitor;
```

```
#
# send the start imaging tag into the data stream and so to the
# binner to start the imaging
#
5      start imaging;

      show status "%%%%%%%%%(script)      Imaging has started, waiting for
the mask to be fully open %%%%%%%%%";

10     # Wait for the mask to be fully open. This is indicated by the binner state changing
from   # Inprogress to ReadyToMove

15     #
      # check that the imaging did start
      #
      _loop = 90;
      while (imaging is not underway)
20     {
          if ( _loop < 0 )
          {
              show status "%%%%%%%%%(script) Imaging did not start
after 90 seconds";
25             choiceNum = message( "DEBUG MESSAGE -- Imaging did not
start after 90 seconds." )
              choices "Abort";
              stop StudyMonitor;
              current setting = false;
30             return;
          }
          _loop = _loop - 1;
          pause 1.0;
35     }

      show status "%%%%%%%%%(script)      Imaging has started, waiting for
the mask to be fully open %%%%%%%%%";

40     # Wait for the mask to be fully open. This is indicated by the binner
      # state changing from Inprogress to ReadyToMove

      _quiet = 0;
      while (imaging is not ready for motion)
45     {
          if ( StudyMonitor is not working)
          {
              show status "%%%%%%%%%(script)      Study Monitor
is not working - waiting for mask to open %%%%%%%%%";
50             stop imaging;
              stop StudyMonitor;
              return;
          }

          if ( imaging is complete )
55     {
```

```

                                show status "%%%%%%%%%(script)      IMAGING IS
COMPLETE - STOPPING 1%%%%%%%%";
                                stop StudyMonitor;
                                return;
5                                }

                                _quiet = _quiet + 1;
                                if (_quiet > 50)
10                                {
                                    show status "%%%%%%%%%(script)      Still waiting
for mask to be fully open %%%%%%%%%";
                                    _quiet = 0;
                                }
                                pause 0.2;
15                                }

                                show status "%%%%%%%%%(script)      Total body mask is fully open,
requesting the start of table motion %%%%%%%%%";
20                                start StudyTB;

                                _quiet = 0;
                                while (imaging is not complete)
25                                {
                                    if ( StudyMonitor is not working)
                                    {
                                        show status "%%%%%%%%%(script)      Study Monitor
is not working - moving gantry %%%%%%%%%";
30                                        stop imaging;
                                        stop StudyMonitor;
                                        return;
                                    }

                                    if ( StudyTB is working )
35                                    {
                                        if ( _quiet > 50 )
                                        {
                                            show status "%%%%%%%%%(script)
40                                            Moving gantry, scanning in progress %%%%%%%%%";
                                            _quiet = 0;
                                        }
                                    }
                                    else
45                                    {
                                        if ( _quiet > 50 )
                                        {
                                            show status "%%%%%%%%%(script)
50                                            Gantry not moving, final curtaining in progress %%%%%%%%%";
                                            _quiet = 0;
                                        }
                                        if ( StudyTB != 1)
                                        {
                                            show warning "Study aborted by user while table was
55                                            moving";
                                            stop StudyMonitor;

```

```

                                return;
                                }
                                }
                                _quiet = _quiet + 1;
5                                pause 0.2;
                                }

                                if ( StudyTB is working )
                                {
10                                # Something is wrong, table should have stopped.
                                show status "%%%%%%%%%(script)          Why didn't the table
                                stop!   Error!!! %%%%%%%%%";
                                choicenum = message( "Longitudinal motion should have stopped, Study
                                Aborted.") choices " OK ";
15                                stop StudyTB;
                                stop StudyMonitor;
                                return;
                                }

20                                if ( StudyTB != 1)
                                {
                                show warning "Study aborted by user while table was moving";
                                stop StudyMonitor;
                                return;
25                                }

                                stop StudyMonitor;
                                current setting = true;
                                show status "%%%%%%%%%(script)          Imaging completed -- Script
30                                tbase.asl done";
                                return;

35                                #      End of the the script

```